

한계라는 이름의 가능성

이창공

20230828

Abstract

우리는 아직 알 수 없는 것이 너무나도 많고, 해결해야 할 문제 또한 너무나도 많다. 결정 불가능성은 소프트웨어에 있어서 우리가 많은 것을 알 수 없는 근본적인 이유이자 한계이다. 이러한 결정 불가능한 행동을 가지는 소프트웨어의 올바름을 증명하는것은 매우 어렵고 중요한 문제이다. 하지만 명확한 한계는 무한한 가능성 또한 제시하였다. 더 정확한 소프트웨어 분석이 가능할 여지는 충분히 있다. 알 수 없다는 것은 수많은 방법론을 낳고 인간의 무궁무진한 가능성을 자극한다.

우리는 아직 알 수 없는 것이 너무나도 많고, 해결해야 할 문제 또한 너무나도 많다. 소프트웨어는 빠른 속도로 발전하여 우리 삶의 구석구석을 차지하고 있다. 하지만 단순히 다음 단어를 예측해 주는 통계 모델이 어떻게 인간처럼 대화할 수 있고 코드까지 생성해주는지, 엄밀한 검사를 거친 소프트웨어가 언제 어디서 결함을 일으킬지, 또 언제쯤 안전이 보장된 소프트웨어를 만들 수 있을지 우리는 아직 알 수 없다.

결정 불가능성은 소프트웨어에 있어서 우리가 많은 것을 알 수 없는 근본적인 이유이자 한계이다. 괴델의 이론과 튜링의 증명은 수학과 알고리즘의 결정 불가능성을 보였다. 인류사의 많은 부분에서 우리는 이와 같은 명확한 한계를 인정하고 싶지 않았다. 프로메테우스의 영원한 징벌과 같이 우리가 올라설 수 있는 어깨를 내준 튜링, 괴델과 같은 거인들은 안타깝게도 계속되는 질타와 고통속에 단명하였다. 그러나 그들의 업적은 곧 인정할 수 밖에 없는 현실로 다가왔다. 튜링의 정지 문제는 소프트웨어의 대부분 행동 및 상태가 결정 불가능 하다는 것을 보였다. 프로그램이 유한한 시간내에 정지 하는지 알 수 없다는 뜻은 곧 해당 프로그램의 메모리 상태나 여러 행동이 결정 불가능 하다는 것을 의미하기 때문이다.

결정 불가능한 행동을 가지는 소프트웨어의 올바름을 증명하는것은 매우 어렵고 중요한 문제이다. 먼저, 소프트웨어의 일부 상태나 행동이 결정 불가능 하다면 온전한 소프트웨어의 분석이 어렵다는 것은 자명하다. 또한 소프트웨어의 수요와 공급이 증가하면서 소프트웨어의 복잡도가 빠른 속도로 증가하고 있다. 결과적으로 소프트웨어 결함의 수 또한 증가하는 추세를 보이고 있다. 단순한 여가부터 다수의 사람의 재산과 생명까지 소프트웨어에 의존하고 있는 현대 사회에서 다수의 결함은 소프트웨어의 신뢰성의 추락을 뜻한다. 따라서 소프트웨어의 올바름을 증명하는 문제는 필수불가결할 것이다.

명확한 한계는 무한한 가능성 또한 제시하였다. 괴델의 결정 불가능성은 이론에서 그치지 않았고 현실 세계로 까지 이어졌다. 양자 스펙트럴 갭 문제는 물리적 세계의 매우 근본적인 것들조차 결정 불가능성을 가질 수 있음을 보였다. 하지만 이러한 사실은 우리의 한계를 제한하지 않았고 오히려 결정 불가능하다는 사실로부터 새로운 방식들을 제안하게 하여 “우리는 알아야만 하며, 알아낼 것”이라는 힐버트의 말을 창의력이란 단어로 도전하게 되었다. 소프트웨어 분석 또한 마찬가지이다. 소프트웨어의 대부분 행동 및 상태가 결정 불가능하다는 사실은 소프트웨어 분석의 발전을 제한하지 못했다. 오히려 더 많은 방법론을 제시하게 하였다. 예를 들어 정적 분석에서 주로 사용되는 요약 해석(Abstract Interpretation)은 메모리의 상태를 가능한 한 안전하게 정의하여 오탐을 포함한 분석을 할 수 있었다. 반면에 동적 분석에서는 프로그램의 제어 구문 조건이나 유전 알고리즘을 이용하여 빠른 속도로 최대한 많은 실행 경로를 탐색하여 미탐을 포함한 완전한 분석을 하는 방법 또한 사용되었다. 최근에는 신경망 기술의 발전을 기반으로 소프트웨어의 통계적인 분석을 하는 여러가지 방법 또한 제시되었다.

더 정확한 소프트웨어 분석이 가능할 여지는 충분히 있다. 기존의 방식은 안전하나 완전하지 못하거나 완전하나 안전하지 못하다. 기존의 여러 방식의 장단점을 결합한다면 기존의 방식들의 한계를 극복할 수 있을 것이다. 여러 기관의 조사에 따르면 분석의 결과로 탐지된 결함이 20%이상인 경우 실용적이지 못하며 대부분의 개발자들이 오탐이 개발팀의 생산성을 저해한다고 답하였다. 따라서 동적 분석의 탐색 방향을 정적 분석의 탐지 결과로 유도하여 탐색 범위를 늘리는 방법과 정적 분석의 오탐을 신경망 기술이나 동적 분석의 탐색으로 줄이는 방법이 고려 될 수 있다. 또한 70년전에 처음 제시되었던 생거 박사의 퍼셉트론 이론이 하드웨어 발전과 함께

신경망이라는 이름으로 현대에 재조명 된 것과 같이 과거에 고려되지 않았던 방식들이 하드웨어의 발전에 따라 재활용되거나 온전히 새로운 방식으로 기존의 방식을 발전시킬 가능성 또한 존재한다.

알 수 없다는 것은 수많은 방법론을 낳고 인간의 무궁무진한 가능성을 자극한다. 아직 우리에게 완벽한 소프트웨어 분석을 할 수 있는 기술은 없다. 하지만 기술의 정확도는 매년 좋아지고 있고, 기업 차원에서 도움이 되고 실제로 사용하는 기술도 등장하고 있다. 우리는 수많은 답을 알 수 없는 문제에 직면하고 고민한다. 하지만 우리는 답을 찾을 것이다. 늘 그러하였듯이.